

SYNCHRO — SETTE

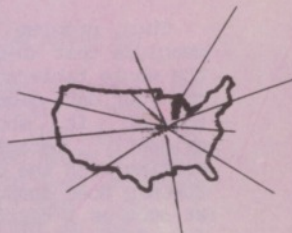
THE SUBSCRIPTION MAGAZINE FOR THE SINCLAIR ZX-81 / TS-1000
MICRO COMPUTERS

VOLUME2 NUMBER1

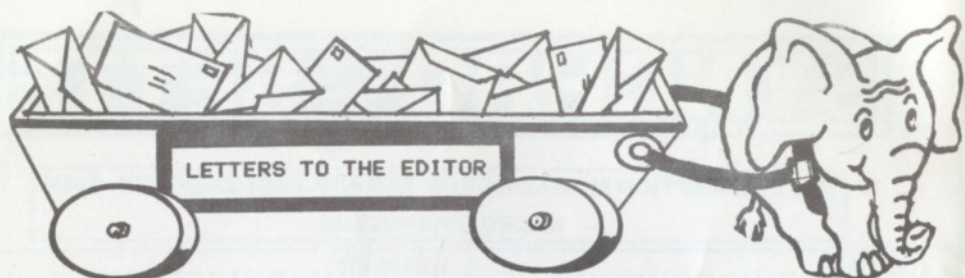


INSIDE THIS ISSUE JAN. 1983

LETTERS TO THE EDITOR	2
HISTORY-THE MARK 1 COMPUTER	6
NUMBER MUNCHER-GAME PROGRAM 7	
THE COMPUTER TUTOR	
SIMULATING PRINT USING	8
1982 CASSETTE PROGRAM INDEX	12
1982 MAGAZINE ARTICLE INDEX	13
JUMP SORT-VISUAL SORT PROGRAM.	14
EDITOR RAMBLINGS	15
HELP YOUR CHILD IN SCHOOL	19



SYNCHRO - SETTE IS PUBLISHED MONTHLY BY : THE S & S COMPANY
388 W. LAKE ST. ADDISON ILL. 60101 (312) 628-8955



Dear Ed,

Your "View of Vu-Calc" in the October issue was helpful to my understanding of this (Timex) program, but as you may have discovered by now, there was a serious flaw in the example which led to an erroneous conclusion. Your example called for formulae in boxes B5 through G5 which required data from box G4. Box G4 also contains a formula rather than data and since Vu-Calc computes by row from left to right starting from row A and then going down row by row, the numeric data required for the formulae in boxes B5 through F5 will not exist and an error report C/5110 results.

This didn't happen with your example, possibly because box G4 contained old data as a result of the step by step way you built up the example, but it will happen when you start with only the raw data.

The Vu-Calc instructions tell you on page 6 to be sure that data required by the formula in a given box, be computed before reaching that box, i.e. in any row above the box on the left of the box in the same row.

This problem also led you to conclude that Vu-Calc didn't compute the whole table at once because some of your data didn't change when you thought it should. That was evidently because it was still working with the old values. In fact Vu-Calc does compute the whole table at once as I verified by scattering formulae all over the table and observing the results change as I changed the input data and then executed the calculate mode. I'm certainly glad it does, because as

you pointed out, it would be very inconvenient if it didn't.

Your example can be restructured to obey the Vu-Calc rules by simply interchanging the rows and columns. Then it works perfectly.

The program also has a simpler way to mix relative and absolute parts in a formula as discovered on page 5 of the instructions. You prefix the parts that are to be absolute with a <\$> when typing in the formula and then choose the relative option as usual to make the other parts relative. This saves having to type in all the formulae individually as you did in your example.

There is one part of the instructions, also on page 5, that I don't understand. The statement is that "you may use up to 40 formulae". That seemed to me to be a terrible limitation and to test it, I put in about 50 formulae. They all worked perfectly - so I don't know if this is a typo (400 allowed) or refers to something I don't understand.

Do you have any information on this?

Sincerely,
Bob Keyser - Encinitas, CA

Read the letter from the following subscriber - Ed.

Dear Ed,

There is a "glaring weakness" in your review of Vu-Calc in the October issue. On page 17 under the heading "Changing Data", paragraph 2

states the data in box 02 has not changed. That is because this column references box G04 which has not yet been calculated.

Page 6 of the instruction sheet provided with the tape, clearly warns against this error. Calculations start at the home square and proceed horizontally along row A, then down to row B, etc.

You missed what I consider the most glaring weakness of the program. The instructions state "You may use up to 40 formulae ---". This apparently means you can press the <F> key to enter a formula up to a maximum of 40 times. After that, you reach an unrecoverable error condition that will not allow you to enter any more formulae. Deleting formulae in the boxes will not allow you to recover from this condition. This happens without warning. If you are using formulae to load data into the boxes, this condition can be reached very quickly.

Other than this, I generally agree with your review (except that the dash between the box co-ordinates in the sample formula is a minus sign). I would like to see more reviews in Synchron-Sette.

Yours Truly,
D.V. Carville
Sperry Univac FSD
New York

Thank you both for your insight in proper use of Vu-Calc and especially for clearing up the data changing format mistake.

Mr. Carville says that after the 40th formula is entered, an unrecoverable error condition occurs where no more formula can be entered and deleting previously entered formula does not help.

Mr. Keyser says that he has entered 50 formulae and they all worked perfectly. Does the 40 maximum refer to individually entered formulae, replicated

formulae or both. Who is right?

In regard to Mr. Carville's comment regarding the dash being a minus sign in the Vu-Calc sample, this to me was one of the most "glaring weaknesses" in the Vu-Calc documentation. The dash looks more like a dot and there is no immediate reference that this dash (dot) is to be a logical operator to perform subtraction. Also, the logical operators for addition, subtraction, multiplication and division should have been shown and explained or at least commented on in the command summary on page 8 of the documentation.

Most people have no idea how many of the subscribers had purchased Vu-Calc and didn't know what to do with it or how to use it until the October article appeared. The most common complaint was the in-ability to understand the documentation and particularly how to enter formulae. The second most common complaint was how to duplicate or replicate formulae.

For those of us who have had experience with Visa-Calc type programs on more expensive computers, Vu-Calc was pretty much second nature and even redundant. All I ask is that we don't make the mistake of assuming the all the readers have the same knowledge we do and write things based on this assumption. I have made this mistake more than once in other areas and will probably make it again. I encourage any chastisement that I deserve from you readers (keep the letters clean, though) - Ed.

Dear Fred (I think he meant Ed)

I received the Sinclair and Mindware printers within a week of each other and I like both. My Sinclair printer also needs a prompt to get the paper to feed (annoying but liveable) and like you, I haven't tried to adjust it yet.

I am now sending for some paper from:

E. Alvarez
PO BOX 1025
Oviedo, FL, 32765

which is supposed to have a whiter background for better contrast (graphics on Sinclair's printer leave a lot to be desired).

Mindware's printer is great. It's just too bad it doesn't use wider paper. The ribbon cartridge from Radio Shack is part # 26-3507. By using packing label tape in the MW-100 for mailing labels, you don't even need to use glue - just lick the back and stick em on.

Your "FILE" OCT/82 program is about the best I've seen. By adding:

```
2375 IF Y$="C" THEN COPY
```

and using the MW-100, it's hard to beat for mailing lists.

I picked up a Diskwasher Competition Joystick and with small wire and one cut inside, it works great on my ZX-81.

You mag is terrific - I just wish they came faster. Keep up the good work.

Jake Breil - Cypress, CA

Dear Jake,

Thanks for the kind words and information. The only problem I have with the print-outs of both printers is that they are extremely hard for the printer to make good proofs from. The grey background of the aluminized paper on the Sinclair printer and the blue ink on the cartridge of the MW-100 are awfully hard for printing proof cameras to pick up.

I find that I am not alone with the problem of having to pull the paper on the Sinclair printer - a few other people have written me with the same situation.

I purchased the Radio Shack

cartridge which has black ink and it printed out very light. It might have been an old ribbon that was partially dried out.

Bob DeLisle of Hammond Indiana sent me some sample copies made with the CAI printer on a white background aluminized paper. The graphics are superb and all characters are easily read. The copy is however, also very light. Whether this is the same type of paper that Alvarez sells, I don't know.

The MW-100 gives the most readable copy I have seen to date - Ed.

Dear Ed,

My main reason for writing is to ask you to put a LPRINT option in your future 'menus'. Maybe there is an easy way to do it but I haven't stumbled on it yet - some of us do have printers, don't forget!

I, for one would like to see more business/home programs & fewer games. The business/home programs are much more fun for me.

D.A. Potts MD - Independence, MO

Dear D.A.,

When the programs that could use print-out were written, most people hadn't even seen a printer yet. We intend to use the print-out capability in future programs.

In the meantime, you are left with 3 options to convert existing programs to print. The first is for listing programs which uses the LLIST command of which you are probably aware of already.

The second is to use the COPY command in a program line. Keep in mind that this command will copy whatever is on the screen - even blank spaces and lines. If you write a routine such as:

```
1000 INPUT AS  
1010 CLS
```


1020 PRINT A\$
1030 COPY

the data you input for A\$ will appear at the top of the screen and the printer will copy it to paper along with all the blank lines below it.

The third procedure, which is probably the most useful, is to use the LPRINT command. This command allows formatting of the print-out to paper. A sample program to demonstrate this would be:

```
10 SCROLL
20 PRINT "ENTER NAME?"
30 INPUT N$
40 SCROLL
50 PRINT "ENTER TAB?"
60 INPUT T
70 SCROLL
80 LPRINT TAB T;N$
90 GOTO 10
```

Play around with this program and try to become familiar with how the print-out can be formatted anywhere on the paper. In almost all programs, it would only require the entry of lines using a format like line 80 uses in this program.

One odd thing I've noticed is that if the last data printed on the screen is at the bottom-most available line position, LPRINT won't work. The computer treats it as a PRINT command and gives a screen full error even though nothing extra is added to the screen. Delete line 70 from the sample and you can observe this

As far as business/home vs. games programs, I'm sure if I changed the ratio, I would here complaints from the other end. We will continue to try to have some of each - Ed.

Dear Ed,

Will use of a mini-drive for the TS-1000 allow one to input data independently of the program? More importantly, do you foresee any possibility of being able to utilize

one set of data in more than one program by being able to access that data from an independent source (another tape recorder or disk drive) or will one be forever destined to duplicate the data each time it is needed for additional programs.

Also, is there any way that you know of where two programs can be chained together so that a common set of data can be used assuming that it is not possible to introduce data independently of a program?

Let me just say that this beginner is ever and eternally grateful to you and all those involved in reaching out to us in such a friendly, scholarly, enthusiastic and helpful way. Although most of us will always be clamoring for "more, more, more!" -- we will all be quite satisfied and grateful for whatever bounties we receive.

Beatrice Spencer
St. Thomas, Virgin Islands

Dear Bea,

I am sorry that I couldn't print your whole letter with all of the questions but in regard to separating data from a program.

There is a utility program that allows data to be stored on tape separate from the program. It uses only 500 bytes of memory and can be used on a 2K machine. It sells for 17.00 (1.50 postage & handling - CA res. add 6% or 6.5% sales tax) and is available from:

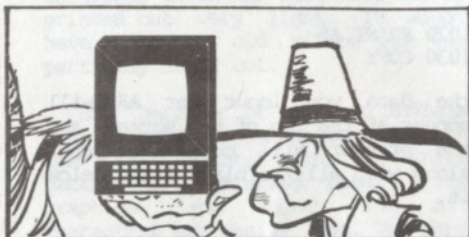
Cosmonics
PO BOX 10358
San Jose, CA, 95157

Otherwise, separate disk drives or recorders won't solve your problem.

As far as chaining programs are concerned, I'm sure it is possible and would require another utility program which I have seen for other

* HISTORY *

THE MARK 1 COMPUTER



Machines that are partly mechanical and partly electrical are "electromechanical". The invention of the ASOC (Automatic Sequence Controlled Calculator) or "Mark 1" was a milestone in electromechanical calculation. It was the joint project of Howard Aiken, professor of applied mathematics at Harvard, and Thomas J. Watson, president of IBM.

Watson was not a designer of machines - he was a salesman and executive - but he had the insight to see the value of Aiken's ideas and to do for him what the chancellor or exchequer was not willing to do for Babbage - underwrite his invention so that it could become a reality.

Aiken wanted to build a general purpose computer that could handle not only his own special mathematical problems but any kind of problem. Watson bought Aiken's ideas and in 1941, let him write his own ticket and gave his four top IBM engineers as helpers on the project.

The work took five years and in 1944 the Mark 1 was formally presented to Harvard University by Watson. It was Babbage's dream come true.

The new machine consisted largely of standard Hollerith counters with a superimposed and specially designed automatic tape sequence control for directing the operations of the machine. The Mark 1 was an electromechanical device 51 feet long and 8 feet high; it had more than 760,000 parts and included 500 miles of wire. This machine weighed 5 tons.

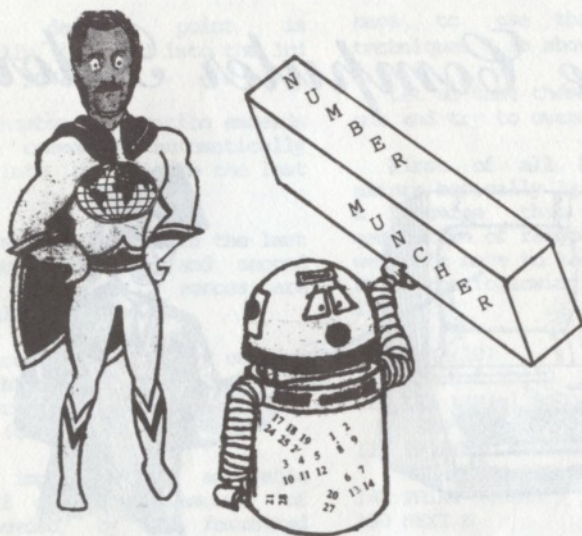
It could carry out any specified sequence of five fundamental operations - addition, subtraction, multiplication, division and reference to tables of results already computed. The input was in the form of punched cards and switch positions. The output was punched into cards or printed by electric typewriters.

Although archaic and bulky by today's standards, one has to admire the ingenuity of the device because there were none of the components used in today's machines such as microprocessors, transistors or even vacuum tubes. The calculating process was done entirely by relays similar to those used in temperature thermostat controls or automobile applications of today.

Needless to say, the process was slow and consumed enormous amounts of electrical energy. The Mark 1 was also very noisy and was said to sound like "a bunch of women, all pounding away at typewriters at the same time" because of the clicking of all the relays.

Despite these drawbacks, the Mark 1, as originally built, could make limited decisions by making comparisons. In 1943, while the first general-purpose electromechanical computer, the Mark 1 was being built, work was being started on the ENIAC, the first electronic computer because it used vacuum tubes. This proved to be the death knell for the electromechanical concept and insured the short life and obsolescence of the Mark 1 and computers of this type.

The Mark 1 had a successor, the Mark 2, which was constructed by the Harvard Laboratory under Aiken's



An excellent and addicting game. Ten inverse numbers between 0 and 9 will appear on the screen at a time. The "muncher" will then proceed from the left to the right of the screen. His direction can be controlled by pressing the <1> key to make him move up and the <0> key to make him move down.

The muncher will make ten passes and after each pass, the new score will be displayed in the center of the screen. Try to get the highest numbers for the highest score.

If the muncher tries to eat the score, the game will be over and a whole bunch of points will be deducted from your score.

This game contains a screen memory-mapping routine that is controlled from lines 40 to 255.

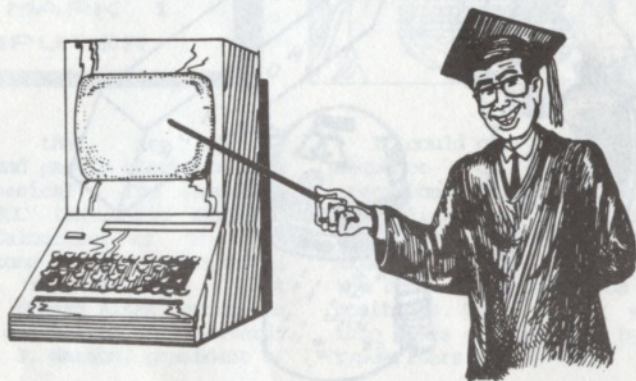
```

1 FAST
2 POKE 16418,0
3 FOR N=1 TO 768
4 PRINT CHR$ 8;
5 NEXT N
6 LET O=255
7 SLOW
40 LET P=(PEEK 16396)+256*(PEE
K 16397)
50 LET S=0
  
```

```

60 LET R=P+133
70 FOR T=1 TO 10
80 FOR Z=1 TO 8
90 LET X=RND*768+P
93 IF X<((RND*400)+P) THEN GOT
O 90
96 IF X>((RND*668)+P) THEN GOT
O 90
100 IF PEEK X=118 THEN GOTO 90
110 POKE X,156+INT (10*RND)
120 NEXT Z
130 FOR Z=1 TO 31
140 IF PEEK (R+Z) <> 8 THEN LET S
=S+PEEK (R+Z)-156
145 IF PEEK (R+Z) =0 THEN GOTO 2
40
150 POKE R+Z,18
155 POKE R+Z-1,22
160 POKE R+Z,8
165 POKE R+Z-1,8
170 IF INKEY$="0" THEN LET R=R+
33
180 IF R>=768+P THEN LET R=R-33
190 IF INKEY$="1" THEN LET R=R-
33
200 IF R<=P THEN LET R=R+33
210 NEXT Z
220 PRINT AT 12,15;S
225 LET O=CODE STR$ S
230 NEXT T
240 CLS
250 PRINT AT 12,6;"YOUR SCORE IS
";S
260 PAUSE 200
270 CLS
280 RUN
  
```

the Computer Tutor



SIMULATING <PRINT USING>

Hello Class! Welcome to the new year. Is everybody ready to resume classwork? Good!

Today's subject will be formatting numbers. All of you, I am sure, have seen printouts of data where numbers were neatly printed in columns and the decimal points all lined up and all the trailing zeroes were added after the decimal point.

Has anyone any idea how this is done? The man in the back has his hand up - yes sir?

The gentleman says that he is not sure how it is done on the Sinclair Computers but on the Radio Shack Computers, a command called PRINT USING is used.

I am glad he brought this up. PRINT USING is a powerful command that allows the user automatic formatting of numbers. A sample program on the TRS-80 would be as follows:

```
10 LET A$ = "###.###.###"  
20 INPUT B  
30 PRINT USING A$;B  
40 GOTO 20
```

If one were to RUN this program on the TRS-80 and ENTER the following values for each input, the values in the right column would be observed:

VALUE	PRINT USING VALUE
1	1.00
12.7	12.70
.005	.01
15796.3	15,796.30
98765432.1	\$98,765,400.00

As you can see, this command easily rounds off any number up to 999999.99 to 2 decimal places. It ignores any numbers more than six digits to the left of the decimal point and more than 2 digits to the right of the decimal point.

So as you can see, it is a very powerful command. This is, however only half of what PRINT USING is doing. The other half involves an automatic TAB function.

FEATURES OF PRINT USING:

- The format in line 10 allows the printing of 10 characters maximum (more or less may be used).

- The decimal point is automatically inserted into the 3rd to the last position.

- If the number in question exceeds 999.99, a comma is automatically inserted into the 7th to the last position.

- If there is no number in the last position or in the last and second to last positions, zeroes are automatically inserted.

- If there are more than 2 numbers to the right of the decimal point, the overage is automatically truncated (chopped off).

- Most important for automatic TABING, if the total amount of spaces needed by the formatted number is less than 10, the PRINT USING command automatically precedes the formatted number with an amount of blank spaces equal to the difference between 10 and the amount of characters of the formatted number.

- The TRS-80, as it stands, only allows 6 digits of accuracy as compared to the 8 digit accuracy of the Sinclair machine. The TRS-80 has the potential for 16 digit accuracy with added commands which we won't discuss here.

If we were allowed to use the PRINT USING command on the Sinclair Computer, we could add a line to the program:

```
25 PRINT TAB 22;"";
```

and all the numbers printed on the screen would have their decimal points lined up in the 30th column and would appear in a nice neat vertical row.

What a shame that this command does not exist for the Sinclair machines.

I see the gentleman in the rear has a tear in his eye. Don't fret, It can still be done - it just requires more work. Don't feel bad, even owners of much more expensive

have to use the same kind of techniques I am about to show you.

Let us take these problems one by one and try to overcome them.

First of all, because we are by nature basically lazy, let us create a routine that will allow the generation of random numbers so that we don't have to keep entering them. ENTER the following program and RUN it:

```
10 DIM A(10)
100 FOR N=1 TO 10
110 LET A(N)=1000000 *
    (RND*RND*RND)
120 IF A(N)<.00000001
    THEN GOTO 110
140 PRINT N,A(N)
150 NEXT N
```

You will observe on the screen 10 random numbers that can be of any amount of digits up to eight and have the decimal point at any possible position. We will now have these numeric values changed into string variables by changing the following lines:

```
10 DIM A$(10,10)
110 LET A$(N)=STR$(1000000 *
    RND*RND*RND*RND)
120 IF VAL A$(N)<.00000001
    THEN GOTO 110
130 IF RND<.5 THEN LET A$(N)=
    A$(N, TO 5)+"" 5 blank spaces ""
135 IF A$(N,5)="" THEN LET
    A$(N,6)=STR$(1+(9* INT
    RND))
140 PRINT N,A$(N)
```

If this conversion is made, the results would appear the same on the screen as if the first program were RUN except that we now get a wider range of differences in the amount of digits after the decimal point and the amount of digits in the number itself.

A big difference has occurred, however. These numbers have been converted to string variables by the STR\$ function and because of this, each number can now be formatted.

We will now truncate each number

to the maximum of 2 positions past the decimal point. Add the following lines to the existing program:

```
160 INPUT B$
170 PRINT AT 0,0;"";
200 FOR N=1 TO 10
210 GOSUB 1000
220 NEXT N
230 STOP
1000 LET A=VAL A$(N)
1010 LET A=INT (100*A+.05)/100
1020 LET A$(N)=STR$ A
1030 PRINT A$(N)
1040 RETURN
```

RUN the program again and after the first set of numbers is displayed, press the ENTER key. You will notice that the second set of numbers will have no more than 2 digits after the decimal point. You will also notice that each number is put to the left of its old corresponding number so that you can easily compare them.

We now have to add any trailing zeroes that are necessary. The following routine will not only do this but will line up the numbers with right margin justification and will even add the separating commas when necessary.

Add or change the following lines:

```
230 INPUT B$
240 PRINT AT 0,0;
300 FOR N = 1 TO 10
310 GOSUB 2000
320 NEXT N
330 STOP
2000 FOR I=10 TO 1 STEP -1
2010 IF A$(N,I)<>" " THEN
    GOTO 2025
2020 NEXT I
2025 LET C$=A$(N,1 TO I)
2030 IF C$(LEN C$-1)="." THEN
    GOTO 2100
2040 IF C$(LEN C$-1)="." THEN
    GOTO 2060
2050 LET C$=C$+".0"
2060 LET C$=C$+"0"
2100 LET A$(N)=" 10 blank spaces "
2105 IF LEN C$>6 THEN LET C$ =
    C$(1 TO LEN C$-6)+"," + C$(LEN
    C$-5 TO LEN C$)
2110 LET A$(N,11-LEN C$ TO 10)=C$
```

```
2120 PRINT TAB 16;A$(N)
2130 RETURN
```

This 2000 subroutine performs the bulk of the formatting. The breakdown of the entire program is as follows:

- Lines 10 to 150 create random numbers up to eight digits long.

Line 110 creates the numbers and line 120 checks to see if the number is too small and if it is, goes back and gets another number.

Lines 130 and 135 are designed to produce a greater amount of numbers with one digit after the decimal point for this example, and line 140 prints each number.

This routine could be replaced by your own such as a data input routine or a routine that created numeric data from other data. The numeric data would then have to be converted to string variables using the STR\$ function as outlined in line 110.

Let us say that the dimensionalized numeric variable A(N) was used in your program. Line 110 would then read:

```
110 LET A$(N)=STR$ A(N)
```

- Lines 160 to 220 and including the subroutine starting with line 1000 truncate the numbers to include no more than 2 digits past the decimal point.

Line 170 sets up the starting position for the first printed number from line 1030.

Subroutine 1000 converts the String variable into a numeric variable in line 1000, truncates the number in line 1010, and then re-converts it back into a string variable and prints it with lines 1020 and 1030.

- Lines 230 to 320 and including the subroutine starting at line 2000 format the finished numbers through string manipulation.

Line 240 sets the format for the starting position of the first printed number from line 2120.

The original string variables (A(N)$) are set to 10 blank spaces. This is accomplished by the first line in the program, line number 10. The $STR$$ function in line 110 stuck a random number with a random amount of digits into each of these string variables with the first digit being in the first character position.

This means that most of the time, some of the right character positions of the string variable were left blank.

For our formatted numbers, we need to accomplish three more objectives;

- 1 - to shift the total number in the string so that the last digit is the last character in the string
- 2 - to add trailing zeroes when necessary
- 3 - to insert a comma when the number exceeds 999.99

The first objective is achieved between lines 2000 to 2025 where a new string variable ($C$$) is created. Because it is not dimensionalized, $\langle C$ \rangle$ cannot have any blank spaces on either side of it and therefore can easily have trailing zeroes and a decimal point tacked on if necessary.

$\langle C$ \rangle$ is created by searching $\langle A$(N) \rangle$ from the right-most position and going to the left until no more blank spaces are encountered. $\langle C$ \rangle$ is then made to be equal to just that part of $\langle A$(N) \rangle$ that has no blank spaces.

Lines 2030 to 2060 add the decimal point and trailing zeroes by searching $\langle C$ \rangle$ to see if it needs any of them.

Since we now have $\langle C$ \rangle$ almost in the format we need it in, we have to do something before we can re-insert it back into $\langle A$(N) \rangle$ and that is that we have to clean out $\langle A$(N) \rangle$ so that it isn't cluttered up with any

part of the old numbers in the wrong positions. This is done by line 2100 that sets it to be equal to 10 blank spaces.

By the way, does any one know why we chose 10 spaces to be the maximum amount in our sample program? Yes, the gentleman in the back has his hand up? I can hardly wait to hear his answer.

Haven't learned much from last year, have you? As usual, his answer is wrong. He said it is because the program was written by a human, and humans invented the decimal system which has a base of ten because man has ten fingers.

What that has to do with anything we're talking about here, I'll never know.

No - the answer is - the ZX/TS computers have a maximum of 8 digits of accuracy for any number that is manipulated through normal mathematical procedures. Well when we convert it to a string and add a decimal point and a comma, we now have 10 total characters in the formatted number.

Anyway, getting back to what we were talking about, all that remains for us to do now before re-inserting $\langle C$ \rangle$ back into $\langle A$(N) \rangle$, is to insert commas into the numbers that exceed 999.99. This is accomplished by line 2105 which checks to see if $\langle C$ \rangle$ is more than 6 characters in length and if it is, inserts the comma in the proper place.

We now come to the final operation before printing the number and that is to re-insert $\langle C$ \rangle$ into $\langle A$(N) \rangle$. You will remember that when $\langle A$(N) \rangle$ originally held a number, the first digit was in the first character position of the string. We must have the exact opposite occur when we re-insert it and that is to have the last digit, even if it is a $\langle 0 \rangle$ occupy the last character position of the string.

In other words, originally the blank spaces were after the number

Index Of 1982 Issues & Cassettes

CASSETTES

PROGRAM	ISSUE	DESCRIPTION
ALIEN-BASE/1K	JUN	GAME
ARTIST/1K	AUG	SPECIAL EFFECTS
BAR-GEN/16K	AUG	BUSINESS
BASECON/16K	APR	COMPUTER MATH
BIORHYTHM/16K	APR	PERSONAL
BIRTHDAY/1K	OCT	PERSONAL
BOUNCING-BOMBS	APR	ARCADE GAME/16K
BULLETIN/16K	AUG/OCT/DEC	BULLETIN BOARD
CHAINSAB/2K	DEC	ARCADE GAME
CHAR-CODE/1K	AUG	UTILITY
CIPHER/16K	JUN	CODE/DECODE WORDS
CLOCK/1K	JUN	UTILITY
FILE/16K	OCT	NAME/ADDRESS FILE
FRENZY/2K	DEC	GAME
INSTRING/1K	APR	UTILITY
INVENTORY/16K	DEC	BUSINESS
LABYRINTH/1K	AUG	ARCADE GAME
MARQUEE-1/1K	JUN	SPECIAL EFFECTS
MARQUEE-2/16K	JUN	BULLETIN BOARD
MESSAGE/1K	AUG	SPECIAL EFFECTS
MICRO-PRO/1K	OCT	WORD PROCESSOR
MILEAGE/2K	DEC	VEHICLE MILEAGE ANALYSIS
MINEFIELD-1/1K	APR	GAME
MINEFIELD-2/1K	APR	GAME
MINI-PRO/16K	OCT	WORD PROCESSOR
MORTGAGE-1/1K	JUN	HOME FINANCE
MORTGAGE-2/16K	JUN	HOME FINANCE
MULTIPLY/1K	AUG	EDUCATIONAL MATH
NAME/ADD/16K	AUG	NAME/ADDRESS FILE
NAMESORT/1K	APR	UTILITY
QUOTE/16K	DEC	BUSINESS
RALLY/1K	APR	GAME
REACTION/16K	OCT	SELF-TEST
RIP-OFF/2K	OCT	SPECIAL EFFECTS
RORSCHACH/1K	APR	GRAPHICS DISPLAY
SCRAMBLED-WORDS	JUN	EDUCATIONAL GAME/16K
SEAS. GREET./2K	DEC	GRAPHICS DISPLAY
SPIDER-DAN/1K	APR	GAME
STAR-LANDER/16K	AUG	ARCADE GAME
SYNCHRO-SMASH	OCT	ARCADE GAME/2K
TWODATES/1K	APR	UTILITY

ARTICLES IN THE 1982 ISSUES OF SYNCHRO-SETTE

THE COMPUTER TUTOR

ISSUE/pg	TITLE	DESCRIPTION
APR/6	READ, DATA & RESTORE	Simulating these commands in Sinclair BASIC
MAY/6	DUMMY SCREEN MEMORY MAP	Memory map in BASIC
JUN/9	VARIABLE CONVERSION	Converting string variables to numeric and back again
JUL/9	WORD PROCESSING	Routines in BASIC
AUG/14	SYNTACTIC SUM	Checking to see if you typed in a program correctly
SEP/9	MOVING GRAPHICS	Routines in BASIC
OCT/8	SORT ROUTINES	Routines in BASIC
NOV/10	STRING MANIPULATION	Routines in BASIC
DEC/13	GRAPHS	Business screen graphing techniques

OTHER ARTICLES

ISSUE/pg	ARTICLE	DESCRIPTION
MAY/4	ARTIFICIAL INTELLIGENCE	Sample programs
JUL/14	CHECKBOOK	Balancing program
MAY/11	COLOR COMPUTER	Sinclair Spectrum news
JUN/14	COLOR COMPUTER	Sinclair Spectrum news
APR/11	PROGRAM BLITZ	Rampack problems
APR/12	CASSETTE RECORDING	Proper methods of SAVEing and LOADING
MAY/14	CASSETTE RECORDING	Proper methods of SAVEing and LOADING
JUL/3	CASSETTE RECORDING	Proper methods of SAVEing and LOADING
NOV/7	CHESS	ZX Master vs. Sargon
JUN/11	CIPHER MACHINE	Coding & Decoding messages
NOV/20	DATES	Utility program
NOV/20	HAPPY HOLIDAYS	Graphics program
NOV/11	HISTORY	First lady of computers
DEC/8	HISTORY	Birth of the modern computer
NOV/16	HOME BUDGET	Software review
JUL/11	IMPACT	Sinclair's market impact
AUG/7	INFLATION	Cost calculating program
DEC/16	PRINTER REVIEW	Sinclair & Mindware
AUG/7	PYRAMID	Graphics program
APR/14	SUPPORTING COMPANIES	Aftermarket products
JUL/14	TITLE	Graphics program
APR/15	USERS' GROUPS	Clubs & associations
MAY/15	USERS' GROUPS	Clubs & associations
JUN/15	USERS' GROUPS	Clubs & Associations
XCT/13	VU-CALC	Software review
MAY/13	ZX-80 VS ZX-81	Comparisons

Other product announcements and reviews are too numerous to list but appear in the Editor Ramblings column.

JUMP SORT



Ever wonder how a sort routine works? Is it hard for you to visualize the swapping of variables?

Here is a program that not only allows you to see the variables change places but also generates the individual variables from the DATA variable in line 110. This simulates the READ, DATA and RESTORE commands found on other BASICS.

To speed up the string variable generation, you might want to add the lines:

```
10 FAST
165 SLOW
```

To slow down the swapping, add the following lines:

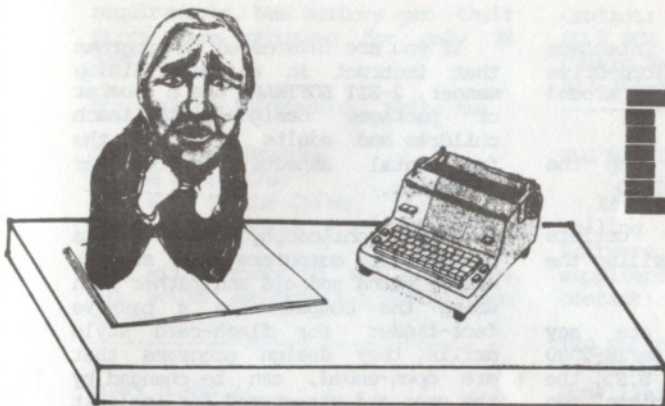
```
2005 FOR C=1 TO 50
2006 NEXT C
2015 FOR C=1 TO 50
2016 NEXT C
```

Try to locate both of the words that are being swapped as they change places on the screen. Line 310 checks to see if one of the variables is larger than the other. It does this by examining the numeric value of the codes of each of the characters in the variable.

If the first variable is larger than the second, then lines 330, 340 and 350 swap them around.

```
100 DIM A$(20,11)
110 LET B$="WATERMELON,BAD JOKE
,TARANTULA,KNEE-HIGH,FOOTBALL,SH
OE,STINKER,CISTERN,POKER,KITE,SL
OP,EYEBALL,LOCKJAW,FLUORESCENT,Z
EBRA,BUFFALO,INTEGER,NORMANDY,AU
TOMOBILE,JUJITSU,"
120 LET M=1
130 LET I=1
140 FOR N=1 TO LEN B$
150 IF B$(N)="," THEN GOSUB 100
```

```
120 LET M=1
130 LET I=1
140 FOR N=1 TO LEN B$
150 IF B$(N)="," THEN GOSUB 100
0
160 NEXT N
170 FOR I=1 TO 19 STEP 2
180 LET J=J+1 J=I+1
190 GGOSUB 2000
200 NEXT I
210 LET P=0
220 LET M=20
230 LET M=INT M/2
240 IF M=0 THEN GOTO 500
250 LET P=P+1
260 PRINT AT 10,20;"PASS NO. ";
P
270 FOR T=1 TO M
280 LET I=T
290 LET J=T+M
300 LET W=0
310 IF A$(I)<A$(J) THEN GOTO 37
0
320 LET W=1
330 LET C$=A$(I)
340 LET A$(I)=A$(J)
350 LET A$(J)=C$
360 GOSUB 2000
370 LET I=J
380 LET J=J+M
390 IF J<21 THEN GOTO 310
400 IF W=0 THEN GOTO 420
410 GOTO 280
420 NEXT T
430 GOTO 230
500 PRINT AT 21,10;"SORT DONE :
::"
510 GOTO 510
500 LET A$(I)=B$(M TO N-1)
1010 LET M=N+1
1030 LET I=I+1
1040 RETURN
2000 LET D$=" "
2010 PRINT AT I,0;A$(I);D$
2020 PRINT AT J,0;A$(J);D$
2030 RETURN
```

EDITOR RAMBLINGS

SYNCHRO-SETTE PRINTED
WITH DAISY WHEEL PRINTER

It seems like only a little while ago we started printing the magazine with the CITHO PRO-WRITER dot matrix printer. This issue is being printed primarily with the Radio Shack Daisy Wheel 2 printer. This simulates typewriter quality print as solid font devices are struck by a miniature hammer against the ribbon and paper. This is, as opposed to the dot matrix method of pins striking the ribbon against the paper.

We hope you like the results.

SPECTRUM MICRO-DRIVES NOT DISK

Consensus of opinion regarding the Spectrum MicroDrives was that they would store data on small circular disks. This is not the case according to a European source.

The storage media is actually a closed-loop cassette tape with an average response time of 3.5 seconds - slower than a floppy disk drive but still much faster than the cassette systems we are used to. Apparently, not all potential purchasers are happy with this situation as evidenced by an excerpt from a letter copy I received:

"With the recent price-cuts by the 'real' micro mfrs., I don't think the SPECTRUM has a snowball's chance here. It will have even LESS of a chance when the Japanese

release their micro-floppy (unless they make the mistake of over-pricing it). The micro-floppy is SPECTRUM's main 'future feature' to entice people to buy NOW. ("There's a fool born ...") It may never be released here. Besides, if their '\$80.00' floppy is of the same quality as their \$100.00 printer, it won't be worth spitting on!

Did I mention, I WON'T be buying a SPECTRUM?"

Strong sentiment, but we shall see. For the price, I don't think the consumer is expecting an IBM PC and from other sources, I have heard the Spectrum is quite a buy for the money.

The Wall Street Journal recently had an article telling the following about the Sinclair computers:

- 100,000 Spectrums have been sold in Britain in 7 months including 50,000 in December.
- 750,000 ZX-81s and TS-1000s were sold in the U.S. last year.
- 16K Spectrums or TS-2000s will sell in the U.S. for around \$150.00.
- 48K Spectrums or TS-2000s will sell in the U.S. for around \$200.00.
- Micro-Drives for the Spectrum/TS-2000 will sell for between \$55.00 to \$70.00 each.

- A special connector or interface will be needed for the Micro-Drive system and will sell for around \$40.00.

- Timex is expected to drop the price of the TS-1000 to \$79.00

- Approximately 14,000 outlets across the U.S. are now selling the TS-1000 and software.

If past records are any indication of the Spectrum/TS-2000 future sales here in the U.S., the gentleman who wrote the "poison pen letter" will indeed learn a lesson about marketing. I'm sure Uncle Clive could care less if he "bought a Spectrum".

The exact date of the debut of the Spectrum/TS-2000 equipment into the U.S. is anybody's guess at this time. We think the computers will come sometime in April or May with the Micro-Drives in July or August of this year.

AMATEUR RADIO

Tom McCarthy is looking for feedback from people who are interested in using the TS/ZX computers with amateur radio equipment - either for RTTY or CW work or for practical software such as keeping a log book, teaching CW, etc.

Tom admits to not being a programming wiz yet but maybe with some help from the editor and interested subscribers, he can write some software and articles to appear in SYNCHRO-SETTE.

If anyone likes this idea and would like to see it get underway, drop a letter to SYNCHRO-SETTE in care of Tom McCarthy/N9DRE along with ideas or software to share. Tom told me to be sure to specify that this is to be a non-profit endeavor.

EDUCATIONAL PROGRAMS

If you are interested in programs that instruct in an entertaining manner, 2-BIT SOFTWARE has a number of packages designed to teach children and adults alike in the fundamental aspects of computer awareness.

Their philosophy involves a belief that computers can stretch minds, young and old and rather than using the computer as a passive fact-finder for flash-card style drills, they design programs that are open-ended, can be changed by the user and structured for implicit learning rather than for a schoolroom approach. Most require only 2K of memory and sell for 9.95.

Subjects of mathematics, probability and theory are covered that encourage analytical and strategic thinking skills with the expected end result being that the user is more comfortable with the computer.

For further information or literature, contact:

2-BIT SOFTWARE
Emerson & Stern
Computer Associates
13674 Boquita Drive
Del Mar, CA, 92014
619-481-3242

* see end

PROGRAM OF THE MONTH CLUB

Original Programs, Phoenix, Arizona has announced the commencement of business of:

The Original Program of the Month Club. They offer a wide selection of programs for the Timex/Sinclair 1000 and ZX-81 on cassette tape. Included are such categories as games, education, family financial, etc. Programs are \$5.99 or less and every sixth program is FREE.

They operate very much like many of the 'book of the month clubs', with limited ongoing purchasing

requirements. New members get their first five programs for only 99 cents.

For further information, write to:

The Original Program
of the Month Club
3763 West Crocus Drive
Phoenix, AZ, 85023

They will respond to all inquiries but would appreciate a SASE. * see below

WINKY BOARD 2

Mr. Russell has done it again!
The WB-2 will now:

- Filter 16K RAMPak noise when SAVEing programs for clean, easy LOADING cassettes.

- DUPLICATE (direct copy) any TS or ZX cassette tape including those special non-LISTable, non-SAVEable cassette programs.

- Set optimum tape volume immediately and precisely with LED indicator lights for any cassette you wish to LOAD.

- Earphone-eavesdrop as you LOAD or DUPLICATE without manipulating cable plugs and cassette volume control.

- Filter electrical noise and interference when LOADING and LOAD those "difficult" cassettes with ease.

- Save time by LOADING and DUPLICATING cassettes SIMULTANEOUSLY.

- Use LED monitors for tape head realignment for those really "difficult" cassettes.

The complete unit assembled and tested is \$24.00 - kit is \$18.00. \$1.00 for earphone, PA residents add 6%, overseas must be in U.S. currency and add \$2.00 shipping (Canada & Mexico no extra shipping charges).

Contact: G. Russell - Electronics
RD 1 BOX 539
Centre Hall, PA, 16828
814-364-1325 eves. * see below

TSG MAILING LISTS

If you would like to purchase a mailing list of Users' Groups, Software suppliers or Hardware suppliers along with TS/ZX owners, contact:

TSG Enterprises
54 Richwood Place
Denville, NJ, 07834

If you would like to get on a mailing list in any of these areas, send them the needed information

* tell em Synchro-Sette sent you!

IT'S A FACT!



A new mortgage plan is available to homeowners and buyers that has lower effective interest rates—sometimes even in single digits. The plan—called the "Fannie Mae" Mortgage Solution—is available on any one- to four-family home on which the existing mortgage is owned by Fannie Mae, the nickname of the Federal National Mortgage Association, which is the largest single source of mortgage money in the country. Under the plan, existing FNMA loans can be "traded in" by a home owner or buyer for a new Fannie Mae loan.

The new loan can be for up to 95 percent of the appraised current market value of the home. A brochure that tells all about the Fannie Mae Mortgage Solution is available free by writing to: Federal National Mortgage Association, Mortgage Solution, 3900 Wisconsin Avenue, N.W., Washington, D.C. 20016.

WHO WERE THEY?

I NOW USE THE
COMPUTER INSTEAD OF
CONSULTING MY
FINANCIAL ADVISORS.



HISTORY CONT.

direction. It was unveiled in 1947, was three times as large and twelve times as fast. It was turned over to the Dahlgren Proving Ground to solve problems in ballistics of guided missiles and supersonic aerodynamics for the U. S. Navy.

Science owes many of its advances to the problems and needs of war. World War 1 saw many new chemical developments and World War 2 accelerated the development of the computer and atomic research.

- Marilyn Buza

LETTERS CONT.

computers but not for the Sinclair. Even if one did exist, I'm not sure that after the programs were merged, the variables would still be intact.

If any readers out there are aware of or have had any experience in this area, please let us know - Ed.

Does anyone have an old ZX-80 manual that they don't need? Please let me know and how much you want for it (cash, trade for software, etc.) - Ed.

EENIE, MEENIE
MINEY AND MOE.



TUTOR CONT.

in the string. Now we want the blank spaces to be at the beginning of the string and proceed the number. Line 2110 accomplishes this.

Line 2120 prints the final formatted number with a TAB command.

This program is easy on the eyes and the 1000 and 2000 subroutines should be useful for any program that requires formatted numeric output involving dollars and cents.

If you have a printer, add a line, 2125 the same as line 2120 but with a LPRINT command.

No homework this time, but experiment with this program and its routines in other programs.

CLASS DISMISSED!

Do you know Presidents talk to the country the way men talk to women? They say, "Trust me, go all the way with me, and everything will be all right." Nine months later you're in trouble.

- Maureen Murphy
on the "Tonight" show

How To Help Your Child With School

Is your child "computer literate?" It's a term that's being used a lot these days to describe a person's basic understanding of computers and programming. In a few years, many people entering the job market without this knowledge may face closed doors. Today, many parents are helping their children to understand and use computers by purchasing a personal computer for the home.

What is a personal computer? Both in the home and in the classroom, a personal computer is a tool for managing and organizing information, as well as a multi-purpose problem solver. Small in size but big in the amount of information they can store, personal computers are making it possible for grade schoolers to work, for example, with geometry concepts years earlier than usual.

As an at-home tutor, a personal computer can teach youngsters language, reading, spelling and mathematical skills, as well as entertain them with game programs. It's also excellent for drilling children in fundamentals.

It's important to look for certain features when choosing a personal computer. Personal computers are fun, but not toys. You should look for the system that best suits your needs and your budget. One new computer, the Timex Sinclair 1000, is only \$99.95. The computer has 2K internal memory, and a 16K RAM pack is available for a suggested retail price of \$49.95. The computer also comes with a 90 day limited warranty from Timex, another important consideration.



As an at-home tutor, a personal computer teaches youngsters languages, reading, spelling and math skills, as well as entertaining them with game programs.

A computer should not be intimidating to you or your family. The Timex Sinclair 1000 offers encouragement, and has been called very "user friendly." Convenient and lightweight, it can be used with any TV set as a monitor.

Computers need software to tell them what to do. Or, you can quickly learn to create your own programs.

A computer for children should be compatible with easy-to-use education programs. The TS1000 has software packages specifically designed for educational use, but, importantly, its instruction manual makes it easy to learn basic programming skills.

It's an important helper for adults as well. A personal computer can help you compute tax information, assess your home energy output, balance your budget, maintain recipe files or catalog possessions. Through the use of a modem, or communications interface device, for low additional cost, the personal computer can give authorized individuals access to large data banks containing financial information, newswires and specific and general research information.



For the 2K Sinclair

GAMES SAMPLER

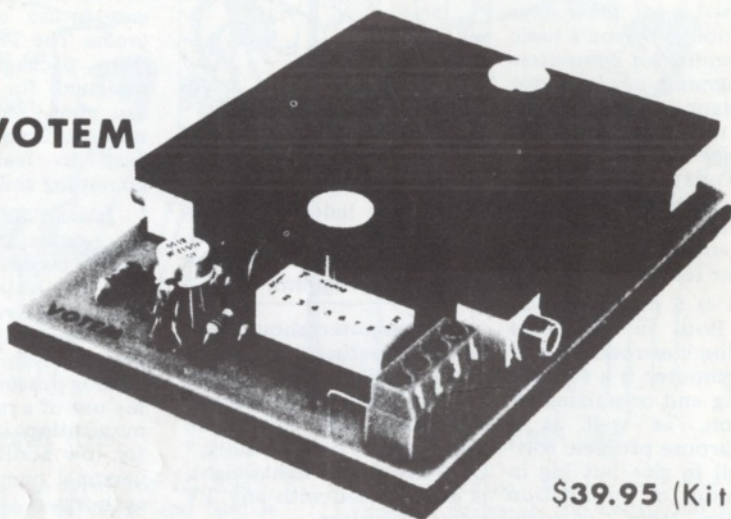
MIND VS. MACHINE

4 different games on each cassette
\$9.95 each

From
Emerson & Stern Associates
13674 Boquita Drive
Del Mar, California 92014

Send for your **FREE** catalog!

VOTEM



\$39.95 (Kit)

ANALOG INTERFACE / TAPE SIGNAL CONDITIONER (for the Timex/Sinclair Computer)

VOTEM is a complete package consisting of hardware and software that enables your computer to measure, display and record "real world" analog signals. Your computer can monitor any physical phenomenon (pressure, light, temperature, etc.) that can be represented by a DC voltage. A probe is provided for air and liquid temperature measurements.

Your computer becomes a "smart" digital voltmeter and thermometer with storage capability. Just think of the possible applications. Use VOTEM and your computer to monitor the temperature in a home energy conservation project to save money and possibly qualify for an energy tax credit.

VOTEM also amplifies and cleans up the tape signal for reliable LOADS. The tape signal conditioner circuit will allow you to LOAD tapes with a lower volume setting on your tape recorder, resulting in less noise and more dependable LOADs. You will be able to LOAD from tapes which would previously not comply.

VOTEM requires no modifications to your computer and does not use the computer's expansion connector, leaving it free for other add-ons such as the memory pack and printer.

At only \$59.95 (assembled and tested), VOTEM is the world's most cost effective analog interface. For an even better bargain the VOTEM kit is only \$39.95. (Requires soldering and appx 2 hrs.) VOTEM comes with a detailed 35-page manual. The manual may be purchased separately for \$5 pp and applied to first purchase of a VOTEM unit. If you are not satisfied with VOTEM return within 15 days for full refund. (Does not apply to kits.) Send check or money order plus \$3 for shipping and handling.

Down East Computers

P.O. Box 3096

Greenville, N.C. 27834

(919) 752-7817 evs.